A COMPARISON OF TWO NONPARAMETRIC DENSITY ESTIMATORS IN THE CONTEXT OF ACTUARIAL LOSS MODEL

MENGJUE TANG

A THESIS

IN THE DEPARTMENT
OF
MATHEMATICS AND STATISTICS

PRESENTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF SCIENCE (MATHEMATICS) AT
CONCORDIA UNIVERSITY
MONTREAL, QUEBEC, CANADA
JULY 2011

CONCORDIA UNIVERSITY
SCHOOL OF GRADUATE STUDIES

This is to verify that the thesis prepared

By: **Mengjue Tang**

Entitled: **A Comparison of Two Nonparametric Density Estimators in the Context of Actuarial Loss Model**

and submitted in partial fulfilment of requirements for the degree of

Master of Science (Mathematics)

complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

Dr. A. Sen _____ Examiner

Dr. D. Sen _____ Examiner

Dr. Y. P. Chaubey _____ Thesis Supervisor

Approved by Dr. J. Garrido _____
Chair of Department or Graduate Program Director

_____2011_____ Dr. B. Lewis _____
Dean of Faculty

ABSTRACT

A Comparison of Two Nonparametric Density Estimators in the Context of Actuarial Loss Model

Mengjue Tang

In this thesis, I will introduce two estimation methods for estimating loss function in actuarial science. Both of them are related to nonparametric density estimation (kernel smoothing). One is derived from transformation based kernel smoothing while the other one is derived from a generalization of Hille's lemma and a perturbation idea that results in a density estimator similar to the kernel density estimator. Both these methods are appropriate for density estimation for non-negative data in general and for actuarial losses in particular. There exist many nonparametric density estimation methods in the literature, but which one should be more appropriate in the context of actuarial losses? I will conduct a simulation study on both of the competing density estimators. The transformation based estimator has been recommended in the literature to be appropriate for the actuarial losses; however, the present study indicates that the new asymmetric kernel density estimator that uses a perturbation idea near zero performs equally well locally as well as globally for many long tailed distributions. The new method is also much simpler to use in practice and hence may be recommended to practitioners in actuarial science.

Acknowledgements

I would like to thank my supervisor Dr. Yogendra P. Chaubey for suggesting me such an interesting topic for my thesis. He had always been willing to help me, in spite of his busy schedule. His encouragement, guidance and support enabled me to finish my thesis. I enjoyed working under his supervision and hope to have an opportunity to work as a PhD student in the near future under his guidance.

I wish to thank Dr. A. Sen and Dr. D. Sen for being my teachers and for accepting to serve on my thesis committee. I appreciate their thoughtful comments on my thesis and for always being there whenever I needed help from them.

I also thank my student colleagues and all my friends; they helped me a lot during my studies and my life in Canada. Particularly, I appreciate Jun Li for sharing some useful information that helped me with completing some computations in my thesis.

Finally I wish to thank my parents for their love and support. I missed them a lot and look forward to be back soon with them after the completion of this thesis.

Contents

List of Tables

List of Figures

Chapter 1

Introduction to Kernel Smoothing

With ever increasing demand of efficient decision-making, as well as extensive use of large database and the rise of data mining, the scope of parametric methods is limited, and this makes the nonparametric density estimation come into play. There are many kinds of nonparametric density estimation methods, such as histogram estimation, kernel density estimation (Parzen-Rosenblatt window method), nearest neighbor estimation and so on. Parzen (1962) proposed fixed bandwidth kernel density estimation, establishing the principle of the bandwidth selection and expanding the utilization of the kernel estimate in mathematical statistics. Rosenblatt (1965) extended this method of estimation to estimation of derivatives of the density. Given a data set, kernel smoothing, or kernel density estimation method can effectively exhibit the data structure without assuming a parametric model. Hence it is widely used nowadays in the areas of social science, medical care, actuarial science and so on. Estimating actuarial loss model is a very interesting and very important problem for all actuaries. Traditional parametric method is computational and with high efficiency but lower uncertainty. Nonparametric method becomes more and more popular because of the development of computer science. If the computation time is no longer a problem for us, we show more

interest to nonparametric estimation (Izenman, 1991).

1.1 General Method of Kernel Density Estimation

1.1.1 Definition

Suppose $X_1, X_2, \cdots\cdots, X_n$ is an independent and identically-distributed sample of a random variable X with its density function $f(x)$, then the kernel density estimator of $f(x)$ is given by

$$\hat{f}(x;h) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x-x_i}{h}\right), \tag{1.1}$$

where K is called the *kernel function,* which is usually a symmetric density function, and h is a smoothing parameter, called *bandwidth.*

Many kinds of kernel functions are available in practice such as Biweight, Triangular, Epanechnikov, Exponential, Uniform, Gaussian and so on, however the standard Gaussian kernel is more popular. Also, since the Gaussian probability density function is infinitely differentiable, this leads to the same property of the probability density function estimator.

It has been observed that the choice of kernel function is not the crucial part in kernel density estimation and any kernel function can guarantee the consistency of the density estimation (Wand and Jones, 1995). However, the bandwidth choice is crucial as it controls the smoothness of the estimator; smaller is the bandwidth, rougher is the estimator. Kernel smoothing is more popular due to its ease of application, mathematical analysis and asymptotic properties, such as strong consistency and asymptotic normality in case of independent

and identically distributed data (Silverman, 1986).

1.1.2 Bandwidth Selection

There are several bandwidth selection criteria that can generally be classified into two categories. One is called " quick and simple " which can find a bandwidth very easily but without too much mathematical computation. The other one is called " hi-tech " which is obviously based on mathematical arguments. There are two methods classified into the first category, that is, Rule-of-thumb and Maximal Smoothing Principle. Basically these two methods are based on AMISE (*the asymptotic mean integrated squared error).* Before we formulate the AMISE, it would be better to explain MISE (*mean integrated squared error*) that is given below:

$$\begin{aligned} MISE\{\hat{f}(x;h)\} &= E\int\{\hat{f}(x;h)-f(x)\}^2dx \\ &= \int\{[Bias(\hat{f}(x;h))]^2+Var(\hat{f}(x;h))\}dx \end{aligned} \tag{1.2}$$

then we can prove that:

$$Bias(\hat{f}(x;h)) = \frac{1}{2}h^2\mu_2(K)f''(x)+o(h^2) \tag{1.3}$$

and

$$Var\{\hat{f}(x;h)\} = \frac{1}{nh}f(x)R(K)+o\left(\frac{1}{nh}\right), \tag{1.4}$$

where

$$\mu_2(K) = \int x^2K(x)dx, \text{ and } R(K) = \int K^2(x)dx.$$

Hence the MSE (*mean squared error)* of $\hat{f}$ is given by,

$$MSE\{\hat{f}(x;h)\} = (nh)^{-1}R(K)f(x)+\frac{1}{4}h^4\mu_2^2(K)(f''(x))^2+o\{(nh)^{-1}+h^4\}. \tag{1.5}$$

MISE is obtained by integrating MSE and therefore we have

$$MISE\{\hat{f}(x;h)\} = AMISE\{\hat{f}(x;h)\}+o\{(nh)^{-1}+h^4\},$$

where

$$AMISE\{\hat{f}(x;h)\} = \frac{1}{nh}R(K) + \frac{1}{4}h^4\mu_2(K)^2R(f'') \tag{1.6}$$

We consider the kernel function K as a probability density function with mean zero, hence we have: $\int K(x)dx = 1$, $\int xK(x)dx = 0$ and $\int x^2K(x) < \infty$.

The optimal bandwidth may be determined by minimizing AMISE. If we differentiate the RHS of the equation (1.6) with respect to h, the optimal bandwidth is as follows:

$$h_{AMISE} = \left[\frac{R(K)}{\mu_2(K)^2R(f'')n}\right]^{1/5}. \tag{1.7}$$

- Rule-of-thumb

  The idea of Rule-of-thumb method was proposed by Deheuvels in 1977. However it is also called Silverman's Rule-of-thumb because it is popularized by Silverman (1986). According to the equation (1.7), what we have to do is try to substitute the unknown f by a reference distribution function. Deheuvels (1977) proposed K as the Gaussian distribution and the standard normal distribution as reference distribution, then the Rule-of-thumb yields the estimator:

  $$\hat{h}_{ROT} = 1.06\hat{\sigma}n^{-1/5} \tag{1.8}$$

  with

  $$R(f'') = \hat{\sigma}^{-5}\frac{3}{8\sqrt{\pi}}, \tag{1.9}$$

  where $\hat{\sigma}^2$ is the sample variance and n is the sample size.

  However there is one problem with the method rule-of-thumb, that is, it is sensitive to outliers. Silverman (1986) suggested a modified estimator which can alleviate this kind of problem. The modified one is as follows:

  $$\hat{h}_{ROT} = 1.06min\left\{\hat{\sigma}, \frac{Q}{1.34}\right\}n^{-1/5}, \tag{1.10}$$

where Q is interquartile range and $Q = X_{[0.75n]} - X_{[0.25n]}$.

Both estimator (1.8) and (1.10) are quite helpful if the true density similar to the normal distribution, but if the true density is far away from the normal distribution, we might get a poor result by using the rule-of-thumb method.

- Maximal smoothing principle

  This principle was introduced by Terrell (1990) that considers finding an upper bound for $R(f'')$. And the estimator according to the *maximal smoothing principle* method is as follows:

$$\hat{h}_{MSP} = 3(35)^{-1/5}\hat{\sigma}\left[\frac{R(K)}{\mu_2(K)^2}\right]^{1/5} n^{-1/5} \tag{1.11}$$

Terrell (1992) strongly advises the use of the“ quick and simple ” method because“ they start with a sort of null hypothesis that there is no structure of interest, and let the data force us to conclude otherwise. ”

The other category of bandwidth selection criterion is “ hi-tech ” and I have introduced it a little bit at the beginning of this section. Since this method is based on mathematical arguments, it appeals to practical applications. Asymptotically, optimal bandwidth selection can be obtained by being switched to minimize MISE$\{\hat{f}(\cdot;h)\}$ if the distribution is continuous. From the equation (1.2), we know that the value of MISE is the summation of Bias square and Variance.

$$MISE\{\hat{f}(x;h)\} = \int \{[Bias(\hat{f}(x;h))]^2 + Var(\hat{f}(x;h))\}dx \tag{1.12}$$

As from the equations (1.3) and (1.4), we know that $Bias(\hat{f}(x;h))$ goes up while $Var(\hat{f}(x;h))$ goes down with the increasing of bandwidth h. That means we should consider both Bias and Variance to make MISE as small as possible so as to achieve the optimal bandwidth. Bias-variance trade-off becomes the key point of this problem.

Next I will introduce three hi-tech bandwidth selection criterions which are commonly used: Cross-validation (including Least squares cross-validation and Biased cross-validation) and Plug-in method.

- Least squares cross-validation

  The least squares cross-validation (LSCV) method was introduced by Rudemo (1982) and Bowman (1984). It is based on the formulae of MISE:

$$MISE\{\hat{f}(x;h)\} = E\int \hat{f}^2(x;h)dx - 2E\int \hat{f}(x;h)f(x)dx + \int f^2(x)dx \tag{1.13}$$

  From the above equation, we can derive an unbiased estimator using the first two terms:

$$LSCV(h) = \int \hat{f}^2(x;h)dx - \frac{2}{n}\sum_{i=1}^{n}\hat{f}_{-i}(X_i;h), \tag{1.14}$$

  where $\hat{f}_{-i}(x;h) = \frac{1}{n-1}\sum_{j\neq i}^{n} K_h(x-X_j)$, $K_h(u) = \frac{1}{h}K(u/h)$.

  The minimizer $\hat{h}_{LSCV}$ of $LSCV(h)$ is then taken to be an estimator for the bandwidth. Hence that we obtain the optimal bandwidth $\hat{h}_{LSCV}$.

- Biased cross-validation

  The biased cross-validation was suggested by Scott and Terrell (1987). It is derived from the formulae of AMISE:

$$AMISE\{\hat{f}(x;h)\} = \frac{1}{nh}R(K) + \frac{1}{4}h^4\mu_2(K)^2R(f''), \tag{1.15}$$

  where $R(K) = \int K^2(x)dx$ and $\mu_2(K) = \int x^2K(x)dx$. Then we have a new estimator: $\widetilde{R(f'')} = \frac{1}{n^2}\sum\sum_{i\neq j}(K_h'' * K_h'')(X_i - X_j)$ which is proposed by Scott and Terrell (1987). The biased cross validation minimizes $AMISE\{\hat{f}(x;h)\}$ with $R(f'')$ replaced by $\widetilde{R(f'')}$. That is, we minimize the following expression as suggested by Scott and Terrell (1987):

$$BCV(h) = \frac{1}{nh}R(K) + \frac{1}{4}h^4\mu_2(K)^2\widetilde{R(f'')}. \tag{1.16}$$

The corresponding bandwidth is denoted by h_{BCV}.

- Plug-in method

The plug-in method was introduced by Sheather and Jones in 1991. Basically it is based on BCV. We can obtain the optimal bandwidth h by differentiating the RHS of the equation (1.6), we get:

$$h_{AMISE} = \left[\frac{R(K)}{\mu_2(K)^2 R(f'')n}\right]^{1/5} \tag{1.17}$$

and then we use a kernel estimator $\hat{R}(f'')$ to replace the $R(f'')$ in equation(1.17), this leads to the plug-in estimator:

$$\hat{h}_{DPI} = \left[\frac{R(K)}{\mu_2(K)^2 \hat{R}(f'')n}\right]^{1/5} \tag{1.18}$$

Usually $\hat{R}(f'')$ is derived from a "pilot " kernel estimate of $f''(x;h)$, which is,

$$\hat{f}''(x;h) = \frac{1}{nh^3}\sum_{i=1}^{n} K''\left(\frac{X_i - x}{h}\right) \tag{1.19}$$

Taking the standard normal kernel $\phi(x)$, this becomes:

$$\hat{R}(f'') = \frac{1}{n^2(\sqrt{2}h)^5}\sum_{i=1}^{n}\sum_{j=1}^{n}\phi^{(4)}\left(\frac{X_i - X_j}{\sqrt{2}h}\right) \tag{1.20}$$

The details about the bandwidth selection criteria can be found in Wand and Jones (1995). And it comes up with a nature question: how to compare those criteria and which one leads to a better result? Actually it is hard to identify which one is better, it depends on in what kind of situations. Basically undersmoothing by LSCV and oversmoothing by BCV and plug-in method show obvious uncertainty of bandwidth selection methods (Loader, 1999). We cannot

fall in love with any one of them. And we can judge which method is better in a different situations using real data example, simulation study and asymptotic analysis. Usually, the performance of LSCV in real data example and simulation study shows a poor result and that makes people feel disappointed in LSCV method. Same thing happens in asymptotic studies because the BCV and Plug-in estimators have faster rate of convergence than that of LSCV (Turlach, 1993). However if we take into account the central processing time, the best bandwidth selection criterions are LSCV or BCV. LSCV and BCV methods take less time and are more efficient than Plug-in method while plug-in provides estimates with a good bias-variance trade-off (Mugdadi and Jetter, 2010). Obviously it is hardly fair to praise any method theoretically. We may have prejudice on some of the bandwidth selectors. The only thing we can do is when we are supposed to analyze a real data set or to do simulation, we can apply different bandwidth selectors to the data and then try to find out the best one by comparing all the possible bandwidths.

1.2 Transformations in Kernel Density Estimation

The kernel estimator gives us a new idea about estimating a density, but it fails to do with some kinds of boundary problems. The kernel estimator performs very well only when the density is quite similar to Gaussian distribution. Otherwise we may get a very poor result. Suppose we have a random variable $X_1, ...X_n$ with its density function $f(x)$. If the domain is on $[0,\infty)$ and we have $f(0) > 0$, then we may find that $\hat{f}(0)$ fails to estimates continuity at boundary $f(0)$. Wand, Marron and Rupert (1991) proposed an idea about general transformation methods which can alleviate this problem more or less. Basically the transformation method is to obtain a new sample whose probability density function is approximately

symmetric as normal distribution. The transformation is given by $Y_i = T(X_i)$ where T is an increasing differentiable function which is defined on the domain of f. Hence that the new transformed kernel density estimator can be written as:

$$\hat{f}(x;h,T) = n^{-1}\sum_{i=1}^{n} K_h\{T(x) - T(X_i)\}T'(x) \tag{1.21}$$

where $K_h(\cdot) = \frac{1}{h}K(\frac{\cdot}{h})$, K is kernel function.

However we still find some problems when we are using the transformation method to do some data analysis. Let's take the log transformation as an example, we get the following transformation estimator: $\hat{f}(x;h) = n^{-1}\sum_{i=1}^{n} K_h\{log(x) - log(X_i)\}\frac{1}{x}$. If we examine the density graph, we notice that "the method could not be fully satisfactory for reducing the boundary bias." (Chaubey et al., 2007). Then we may have to find out other methods to deal with the boundary problem.

" How to choose the transformation T " is another important question. It quite depends on the shape of the original density function. In the next chapter, I will discuss about it in detail.

Chapter 2

Semi-parametric Transformation Kernel Smoothing of Actuarial Loss Function

2.1 Actuarial Loss Model

A loss distribution is a distribution of a positive random variable that has long tail in order to allow large claims. According to these properties, actuaries will use a method which has positive domain and is good at estimating the density at the tails. Actually, it is equally for actuaries to estimate all the possible losses (small losses, medium losses and large losses) in a loss distribution. However large losses may suggest us to reconsider insurance contract, which makes people pay more attention to the density tails. Traditionally, we use parametric models such as log-normal distribution or Pareto distribution or some kinds of mixture of them which have relatively heavy tail. We have to notice that no matter what kind of loss model we choose, it is only an approximation of real data. A loss model is a probability distribution depicting the probability for the number of dollars paid on an insurance claims. Naturally the corresponding random variable is non-negative. If we use parametric

models, we have to compare the competing models so as to figure out which one is the best and simplest. In case standard models don't fit well, things get complicated for actuaries. Nonparametric method may be useful in such cases where we have large insurance portfolios (Klugman et al., 2008). If nonparametric smoothing method would help solving the problems for actuaries, it will be widely used because of its lower uncertainty and simplicity. However classical kernel smoothing fails to estimate the density tails. By noticing that, a method using in the paper "Kernel density estimation of actuarial loss functions" (Bolance et al., 2002) comes up, which is a slightly adjusted version of the semi-parametric transformation method of Wand et al. (1991).

2.2 The Shifted Power Transformation

In the first chapter, we already know the very basic knowledge about transformation method. In this section, we will mainly discuss about one family of transformations which is called shifted power transformation. Basically it is an extension of Box-Cox transformation. With respect to the paper Bolance et al. (2002), the semi-parametric transformed method behaves very well in estimating actuarial loss functions because the transformation gives a symmetric distribution. There are three reasons for doing so. Firstly, it is quite useful to deal with the problem of heavy tail. Secondly, it is reasonable to use a simple rule-of-thumb bandwidth selection criterion when estimating the density which can make things easier. Lastly, it more or less alleviates the boundary problem.

The shifted power transformation modified by Wand et al. (1991) is

$$y = g_{\lambda}(x) = \begin{cases} (x+\lambda_1)^{\lambda_2} & if \quad \lambda_2 \neq 0, \\ ln(x+\lambda_1) & if \quad \lambda_2 = 0, \end{cases} \tag{2.1}$$

where $\lambda_1 > -min(X_1,...,X_n)$ and $\lambda_2 < 1$. (λ_1,λ_2) are transformed parameters. According to the equation (1.21), we can obtain the transformed density as follows:

$$f_y(y,\lambda) = f\{g_\lambda^{-1}(y)\}(g_\lambda^{-1})'(y) \tag{2.2}$$

If we use the standard kernel density estimator, then the transformed density f_y can be estimated by the following estimator:

$$\hat{f}(x,\lambda) = g'_\lambda(x)n^{-1}\sum_{i=1}^{n} K_b(g_\lambda(x) - g_\lambda(X_i)) \tag{2.3}$$

To estimate loss function using semi-parametric transformation method, we have to focus on the transformation parameters selection and the bandwidth selection. Obviously they are crucial parts of estimating loss models.

- Transformation parameters selection

  In order to find the optimum value of λ, we minimize the MISE given in equation (2.3). Substituting the optimal bandwidth h given in the equation (1.8) into the equation (1.6), the MISE of the equation (2.3) becomes:

  $$MISE\{\hat{f}(x,\lambda)\} = \frac{5}{4}[\mu_2(K)R^2(K)]^{2/5}R(f_y'')^{1/5}n^{-4/5} \tag{2.4}$$

  where

  $$R(f_y'') = \int_{-\infty}^{+\infty}[f_y''(y,\lambda)]^2 dy \tag{2.5}$$

  Minimizing the quantity in the RHS of the equation (2.4) is equivalent to minimize the quantity in the equation (2.5).

  In order to estimate $R(f_y'')$, Hall and Marron (1987) introduced the following estimator:

  $$\hat{R}(f_y'') = n^{-1}(n-1)^{-1}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} c^{-5}K \times K\{c^{-1}(Y_i - Y_j)\} \tag{2.6}$$

Suppose that f_y is a normal distribution, then the bandwidth c for the above estimator can be estimated by the following equation:

$$\hat{c} = \hat{\sigma}_x (21/40\sqrt{2}n^2)^{1/13} \tag{2.7}$$

where $\hat{\sigma}_x = \sqrt{n^{-1}\sum_{i=1}^{n}(Y_i - \overline{Y})^2}$ (Park and Marron (1990)).

After determining the optimal transformation parameters, we are going to select the optimal bandwidth which will be used in the transformed estimator (2.3).

- Bandwidth selection

  According to the bandwidth selection criteria which was introduced in Section 1.1.2, we can simply use the rule-of-thumb method here since the true density is very similar to normal distribution. This "quick and simple" method is very attractive for the actuaries because it cause less time. Bolance used shifted power transformation to make the density zero skewness. It seems reasonable to just apply the rule-of-thumb to estimate the bandwidth. Then the following bandwidth estimator is:

$$\hat{b} = 1.059\hat{\sigma}_x n^{-1/5} \tag{2.8}$$

Using the estimators for transformation parameters and bandwidth, we denote the corresponding transformation estimator for equation (2.3) as $\hat{f}(x, \hat{\lambda}, \hat{b})$.

In the chapter 4, there is a simulation study to check if the semi-parametric transformation kernel smoothing suitable for estimating actuarial loss models, as compared to a new non-parametric density estimation method which will be introduced in the next chapter.

Chapter 3

A New Smooth Density Estimator

Compared to the other kernel density estimators, the one I introduced in the Chapter 2 seems more efficient. It is because the semi-transformation method more or less alleviates the boundary problem which is also important in estimating some models including loss models. However, it is not the only way to deal with the boundary problem and we are more interested in finding a new estimator which is similar to kernel density estimator and without data transformation. In order to simplify the computational process, Chaubey et al. (2007) proposed a new smooth density estimator for non-negative random variables. The new estimator is based on two ideas: one is generalization of Hille's lemma and the other is perturbation idea. If we combine these two ideas, we can find it quite helpful to alleviate the boundary problem. And we are expecting it to be also efficient to the heavy tail estimation. I will compare the two methods which are both efficient in dealing with the boundary and to find which one is relatively better in estimating loss model (especially in estimating the tails) in the next chapter. By performing simulation study on these two methods, we can easily find which one is more suitable that cause less error for loss model theoretically. In the next section I will introduce the general idea about the new smooth density estimator.

3.1 General Idea of the New Estimator

When we are estimating a density, we are always trying to look for a more efficient and easier way. A new estimator derives from Hille's lemma gives us a different idea to estimate non-negative random variables.

Lemma 1 (Hille's Lemma)

Let $u(x)$ be any bounded and continuous function on $\mathbb{R}^+$. Then

$$e^{-\lambda x}\sum_{k\geq 0}\frac{(\lambda x)^k}{k!}u\left(\frac{k}{\lambda}\right)\rightarrow u(x), as \quad \lambda\rightarrow\infty, \tag{3.1}$$

uniformly in any finite interval in $\mathbb{R}^+$, where λ is a given non-negative constant.

The above lemma is introduced in Feller (1965) that had been used by Chaubey and Sen (1996) for suggesting smooth density estimators based on Poisson probabilities. Later, Chaubey et al. (2007) used a generalization of the above lemma in proposing a new kernel type density estimator that will be outlined now.

Lemma 2 (Generalization of Hille's lemma)

Let $u(x)$ be any bounded and continuous function on $\mathbb{R}^+$and $G_{x,n}$ be any family of distribution with mean $\mu_n(x)$ and variance $h_n^2(x)$. Then

$$\tilde{u}(x)=\int_{-\infty}^{\infty}u(t)dG_{x,n}(t)\rightarrow u(x). \tag{3.2}$$

uniformly in any finite interval in $\mathbb{R}^+$, where $\mu_n(x)\rightarrow x$ and $h_n(x)\rightarrow 0$.

Next suppose we have a random variable $X_1, X_2, ..., X_n$, the empirical distribution function F_n is defined as follows:

$$F_n(x)=\frac{1}{n}\sum_{i=1}^{n}I(X_i\leq x), x\geq 0 \tag{3.3}$$

Now if we apply lemma 2 to estimate a distribution function by substituting $u(x)$ by the empirical distribution function $F_n(x)$, we obtain the following equation

$$\tilde{F}_n(x) = \int_{-\infty}^{\infty} F_n(t) dG_{x,n}(t) \to F_n(x). \tag{3.4}$$

For a non-negative random variable X, suppose the cumulative distribution function is F, then the survival function S is defined as

$$S(x) = 1 - F(x) \tag{3.5}$$

Then the equations (3.4) and (3.5) motivate the following estimator of $F(x)$

$$F_n^+(x) = 1 - \frac{1}{n} \sum_{i=1}^{n} Q_{v_n}\left(\frac{X_i}{x}\right) \tag{3.6}$$

$Q_v(x)$ is a distribution on positive domain with mean 1 and variance v^2, where $v_n \to 0$ as $n \to \infty$.

Obviously, if we take the derivative of $F_n^+(x)$, we can obtain the estimator of density function $f(x)$ as follows

$$\frac{d}{dx}(F_n^+(x)) = \frac{1}{nx^2} \sum_{i=1}^{n} X_i q_{v_n}\left(\frac{X_i}{x}\right) \tag{3.7}$$

In the above equation, we note that x cannot be zero, hence we modify the above estimator by just simply applying the idea of perturbation, that gives the following estimator:

$$f_n^+(x) = \frac{1}{n(x+\varepsilon_n)^2} \sum_{i=1}^{n} X_i q_{v_n}\left(\frac{X_i}{x+\varepsilon_n}\right), x \geq 0 \tag{3.8}$$

where $\varepsilon_n \to 0$ at an appropriate rate as $n \to \infty$.

In order to deal with the boundary problem better, Chaubey et al. (2007) suggested a corrected version of the estimator (3.8). That is

$$f_n^*(x) = \frac{f_n^+(x)}{c_n} \tag{3.9}$$

where c_n is a constant and has the value of

$$c_n = \frac{1}{n}\sum_{i=1}^{n} Q_{v_n}\left(\frac{X_i}{\varepsilon_n}\right) \tag{3.10}$$

Therefore our new estimator is as follows:

$$f_n^*(x) = \frac{\frac{1}{(x+\varepsilon_n)^2}\sum X_i q_{v_n}(\frac{X_i}{x+\varepsilon_n})}{\sum Q_{v_n}(\frac{X_i}{\varepsilon_n})} \tag{3.11}$$

When we study the asymptotic properties of the new estimator, we find that the new estimator is strongly consistent which is a very important property in estimation theory. Also the new estimator has the asymptotically normal distribution. By proving all the asymptotic theorems, we can conclude that the new estimator is reasonable and will work quite well in practice. All the proofs are given in the paper of Chaubey et al. (2007).

It seems that both of the estimators work quite good theoretically, especially in dealing with the boundary problem. However when estimating the loss model, it is more important to estimate the tail. So in the next chapter, I will compare the two estimators by simulating some loss models and find how different they are.

After introducing the new estimator, we have to decide: how to choose the smoothing parameters of the equation (3.11)? For the new estimator (3.11), we usually take the function $Q_{v_n}(.)$ to be the Gamma distribution function with shape $1/v^2$ and scale v^2. Then what we have to do is to decide how to obtain the optimal values of v and ε_n. We already know the new estimator is quite similar to the kernel smoothing, so we can simply apply the bandwidth selection criterions in the section 1.1.2 to the new estimator. And we may find that those criterions work quite good for the new smooth density estimator.

3.2 Selection of the Smoothing Parameters

According to the asymptotic studies and previous experience, BCV method is more preferable because of its relatively faster convergence and its more efficiency. We can use the BCV criterion to determine what is the optimal smoothing parameters. According to the section 1.1.2, the BCV method is based on AMISE. Firstly we find the values of Bias and Variance of estimator (3.9) which is not difficult. The Bias and the Variance of estimator (3.11) can be obtained easily given that the Bias and the Variance of estimator (3.9). We need to use the definition of Bias and Variance, together with Taylor's expansion and the asymptotic properties of the equation (3.9). More details can be found in the paper Chaubey et al. (2007). Hence that we can obtain the following information about the new estimator (3.11):

$$Bias[f_n^*(x)] = \frac{xv_n^2+\varepsilon_n}{c_n} f'(x) + o(v_n^2+\varepsilon_n), \quad v_n^2 \to 0 \quad and \quad \varepsilon_n \to 0. \tag{3.12}$$

$$Var[f_n^*(x)] = \frac{I_2(q)f(x)}{nc_n^2 v_n(x+\varepsilon_n)} + o((nv_n)^{-1}), \quad v_n \to 0, \quad \varepsilon_n \to 0 \quad and \quad nv_n \to \infty \tag{3.13}$$

Where $I_2(q) = 1/\sqrt{4\pi}$.

MSE shows up when we combine the Bias and the Variance

$$MSE[f_n^*(x)] = \frac{I_2(q)f(x)}{nc_n^2 v_n(x+\varepsilon_n)} + \left[\frac{xv_n^2+\varepsilon_n}{c_n} f'(x)\right]^2 + o(v_n^2+\varepsilon_n) + o((nv_n)^{-1}) \tag{3.14}$$

After taking integration of MSE, we get the value of MISE as follows:

$$MISE[f_n^*(x)] = \frac{I_2(q)}{nc_n^2 v_n}\int_0^\infty \frac{f(x)}{x+\varepsilon_n}dx + \int_0^\infty \left[\frac{xv_n^2+\varepsilon_n}{c_n} f'(x)\right]^2 dx + o(v_n^2+\varepsilon_n) + o((nv_n)^{-1}) \tag{3.15}$$

With respect to the asymptotic property, we know that the leading term of MISE is so called AMISE which is the key to the parameter selection. The leading term of MISE is:

$$AMISE[f_n^*(x)] = \frac{I_2(q)}{nc_n^2 v_n}\int_0^\infty \frac{f(x)}{x+\varepsilon_n}dx + \int_0^\infty \left[\frac{xv_n^2+\varepsilon_n}{c_n}f'(x)\right]^2 dx \tag{3.16}$$

Recalling the method BCV in the chapter 1, we get the BCV by substituting $f(x)$ by $f_n^*(x)$ in the equation (3.16):

$$BCV = \frac{I_2(q)}{nc_n^2 v_n}\int_0^\infty \frac{f_n^*(x)}{x+\varepsilon_n}dx + \int_0^\infty \left[\frac{xv_n^2+\varepsilon_n}{c_n}f_n^{*'}(x)\right]^2 dx \tag{3.17}$$

The pair of (v_n, ε_n) which can minimize the above BCV function (3.17) is our best choice of parameters. Let's denote it as $(\hat{v}_n, \hat{\varepsilon}_n)$. By knowing the optimal estimator for smoothing parameters and bandwidth, we denote the corresponding new estimator for equation (2.3) as $\hat{f}(x, \hat{v_n}, \hat{\varepsilon_n})$. Next Chapter I will compare the performance of the two estimators $\hat{f}(x, \hat{\lambda}, \hat{b})$ and $\hat{f}(x, \hat{v_n}, \hat{\varepsilon_n})$.

Chapter 4

A Simulation Study

4.1 Introduction

In order to compare the two estimators in Chapter 2 and Chapter 3, I take advantage of simulation method. Similar to the paper of Bolance et al. (2002), I also consider 6 loss models and compute two measures of departure from the target distribution. These are based on the values of ISE (*integrated squared error)* and WISE (*weighted integrated squared error)* that will be defined later in this section. For WISE, the weights are proportional to x^2 as this measure puts smaller weights for densities closer to zero, i.e. at tails of the distribution. Still these measures are appropriate for global comparison of the density estimators. In order to judge the estimators locally we would like to compare expected squared errors in specific regions of supports, for example in the tail of the distribution. In actuarial science, we use some densities which allow skewed and heavy tail distributions such as Pareto or lognormal or some kinds of mixture of them. In order to figure out this problem more comprehensive, the six loss models we generate are: a Weibull distribution with parameter 1.5, a lognormal distribution with parameters (0,0.5), a lognormal distribution with parameters (0,1), a mixture

model of 70% lognormal distribution with parameters (0,1) and 30% Pareto distribution with parameter 1, a mixture model of 40% lognormal distribution with parameters (0,1) and 60% Pareto distribution with parameter 1, a mixture model of 10% lognormal distribution with parameters (0,1) and 90% Pareto distribution with parameter 1. The domain for each model is on positive.

I generate n (n=100, n=200, n=1000) random numbers from the above six models, then examine the performance of the two estimator by comparing the ISE and WISE. Each simulation will be conducted replicated 100 times in order to get a more accurate result. The basic idea of calculating ISE is as follows:

$$ISE = \int_{-\infty}^{+\infty} \{\hat{f}(x) - f(x)\}^2 dx \tag{4.1}$$

As we can see, the ISE is the squared value of the distance between the estimated density and the simulated density, integrated over the support of the distribution and WISE is similarly interpreted, where the weights are x^2, thus

$$WISE = \int_{-\infty}^{+\infty} \{\hat{f}(x) - f(x)\}^2 x^2 dx. \tag{4.2}$$

To compare the two estimators based on simulation, we could use simulated averages of ISE and WISE, however, we have considered the following two measures as goodness of the estimators as in Bolance et al. (2002):

$$D_1 = \frac{1}{n} \sum_{i=1}^{n} \{\hat{f}(X_i) - f(X_i)\}^2, \tag{4.3}$$

$$D_2 = \frac{1}{n} \sum_{i=1}^{n} \{\hat{f}(X_i) - f(X_i)\}^2 X_i^2. \tag{4.4}$$

In order to see the distribution of these divergence measures, we repeat the sampling 100 times for each sample size. Even though this provides a limited study, it does give a relative comparison of the two estimators based on the same set. A small number of replications is chosen, specially for local comparison as the computing time required becomes enormous for larger replications. In the next section I provide, the mean, median and sd (standard deviation) of simulated values D_1 and D_2.

4.2 Global Comparison of the Two Estimators

Here are the results of global comparison for the two estimations. Smaller value of D_1 and D_2 indicate better performance of the corresponding estimation method.

Table 4.1: Transformation estimator with Weibull distribution (c=1.5)

		$n = 100$	$n = 200$	$n = 1000$
ISE	mean	0.01344	0.00737	0.00191
	median	0.00990	0.00530	0.00157
	sd	0.01091	0.00595	0.00145
WISE	mean	0.00680	0.00371	0.00091
	median	0.00532	0.00281	0.00082
	sd	0.00536	0.00311	0.00049

Table 4.2: New estimator with Weibull distribution (c=1.5)

		$n = 100$	$n = 200$	$n = 1000$
ISE	mean	0.01053	0.00694	0.00257
	median	0.00757	0.00566	0.00181
	sd	0.00892	0.00499	0.00239
WISE	mean	0.00591	0.00394	0.00142
	median	0.00442	0.00319	0.00101
	sd	0.00478	0.00279	0.00142

According to these summary statistics presented in the tables, the transformation method performs pretty much the same as the new estimator for the first three models. When n is large enough, the D_1 and D_2 values for the transformation method are a little smaller than those of the new estimator. However the new estimator seems better than the transformation method for the other three mixture models as based smaller values of D_1 and D_2 for the new estimator. Usually loss models have long tails in order to allow large claims. If we sketch the graphs of each simulated density, we can find that which model is more suitable to describe a loss model. The simulated densities of all these six models are presented in Figure 4.1.

Table 4.3: Transformation estimator with lognormal distribution ($\sigma = 0.5$)

		$n = 100$	$n = 200$	$n = 1000$
ISE	mean	0.00617	0.00418	0.00141
	median	0.00549	0.00350	0.00121
	sd	0.00444	0.00277	0.00094
WISE	mean	0.00419	0.00290	0.00103
	median	0.00364	0.00239	0.00088
	sd	0.00298	0.00202	0.00075

Table 4.4: New estimator with lognormal distribution ($\sigma = 0.5$)

		$n = 100$	$n = 200$	$n = 1000$
ISE	mean	0.00857	0.00538	0.00166
	median	0.00758	0.00487	0.00124
	sd	0.00625	0.00351	0.00128
WISE	mean	0.00638	0.00396	0.00125
	median	0.00586	0.00352	0.00095
	sd	0.00467	0.00266	0.00103

Table 4.5: Transformation estimator with lognormal distribution ($\sigma = 1$)

		$n = 100$	$n = 200$	$n = 1000$
ISE	mean	0.00438	0.00271	0.00081
	median	0.00368	0.00215	0.00068
	sd	0.00421	0.00207	0.00055
WISE	mean	0.00209	0.00134	0.00042
	median	0.00176	0.00102	0.00033
	sd	0.00192	0.00105	0.00031

Table 4.6: New estimator with lognormal distribution ($\sigma = 1$)

		$n = 100$	$n = 200$	$n = 1000$
ISE	mean	0.00481	0.00289	0.00084
	median	0.00400	0.00247	0.00070
	sd	0.00369	0.00190	0.00062
WISE	mean	0.00247	0.00148	0.00044
	median	0.00209	0.00125	0.00034
	sd	0.00197	0.00010	0.00036

Table 4.7: Transformation estimator with mixture model 30% Pareto + 70% lognormal

		$n = 100$	$n = 200$	$n = 1000$
ISE	mean	0.02917	0.02244	0.02659
	median	0.00728	0.00427	0.00178
	sd	0.09478	0.09789	0.19054
WISE	mean	0.00994	0.00736	0.00816
	median	0.00329	0.00195	0.00069
	sd	0.02839	0.02941	0.05716

Table 4.8: New estimator with mixture model 30% Pareto + 70% lognormal

		$n = 100$	$n = 200$	$n = 1000$
ISE	mean	0.00610	0.00371	0.00142
	median	0.00508	0.00332	0.00111
	sd	0.00468	0.00244	0.00097
WISE	mean	0.00296	0.00177	0.00068
	median	0.00252	0.00158	0.00053
	sd	0.00214	0.00112	0.00048

Table 4.9: Transformation estimator with mixture model 60% Pareto + 40% lognormal

		$n = 100$	$n = 200$	$n = 1000$
ISE	mean	0.31726	0.30106	0.06680
	median	0.01245	0.00865	0.00349
	sd	2.75066	2.38577	0.38046
WISE	mean	0.18993	0.18068	0.04003
	median	0.00724	0.00529	0.00203
	sd	1.65005	1.43128	0.22826

Table 4.10: New estimator with mixture model 60% Pareto + 40% lognormal

		$n = 100$	$n = 200$	$n = 1000$
ISE	mean	0.00802	0.00317	0.00137
	median	0.00429	0.00305	0.00123
	sd	0.00947	0.00201	0.00084
WISE	mean	0.00262	0.00099	0.00042
	median	0.00132	0.00092	0.00037
	sd	0.00318	0.00063	0.00025

Table 4.11: Transformation estimator with mixture model 90% Pareto + 10% lognormal

		$n = 100$	$n = 200$	$n = 1000$
ISE	mean	0.88790	0.40273	0.08531
	median	0.01966	0.01221	0.00671
	sd	7.02564	2.16752	0.31053
WISE	mean	0.58946	0.44495	0.02704
	median	0.01436	0.00809	0.00361
	sd	3.45938	2.51473	0.10225

Table 4.12: New estimator with mixture model 90% Pareto + 10% lognormal

		$n = 100$	$n = 200$	$n = 1000$
ISE	mean	0.01746	0.00467	0.00175
	median	0.00776	0.00310	0.00156
	sd	0.02113	0.00625	0.00091
WISE	mean	0.00174	0.00045	0.00017
	median	0.00077	0.00030	0.00016
	sd	0.00213	0.00062	0.00009

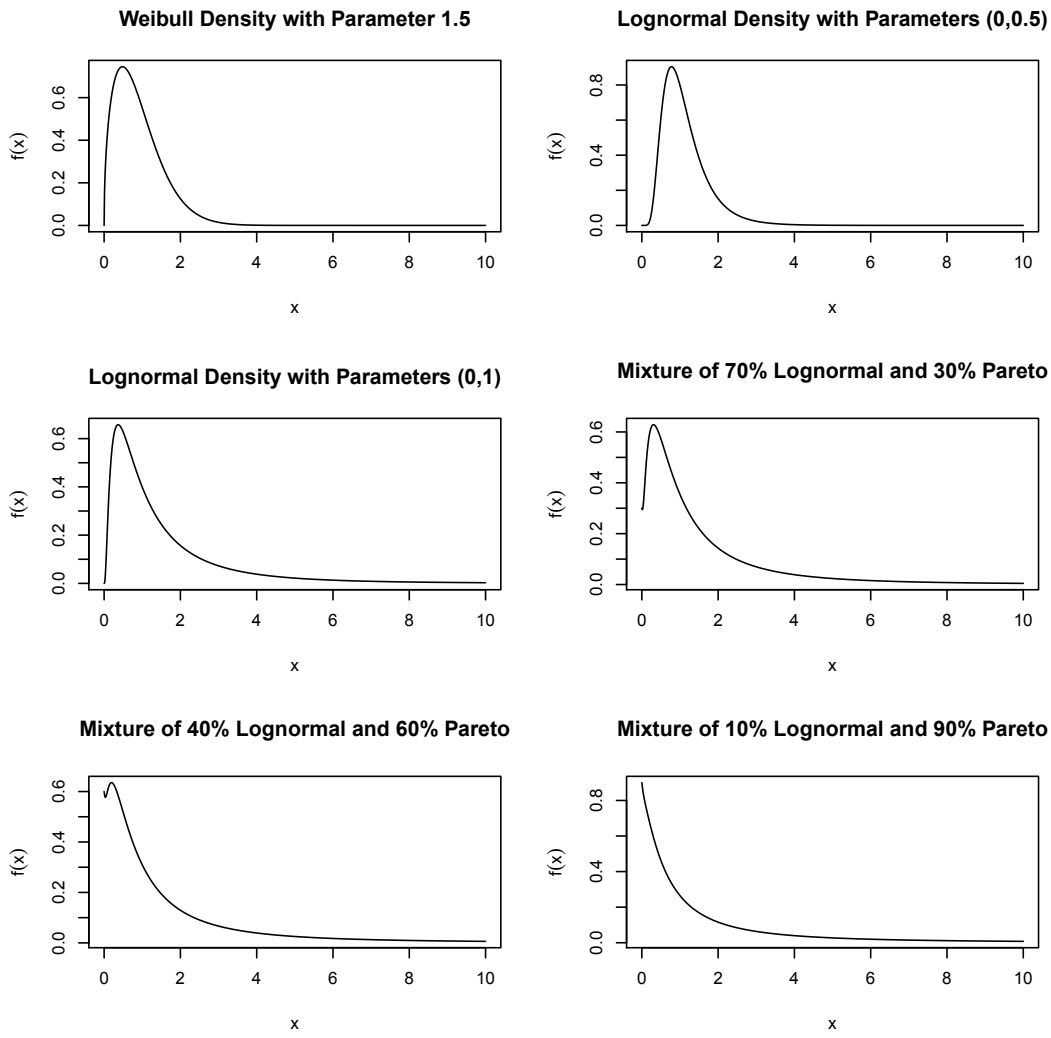

Figure 4.1: Simulated densities of the six loss models

Usually actuaries prefers to use the mixture model such as the mixture of 30% Pareto and 70% lognormal when they want to simulate a loss model. Hence if we focus on the table 4 and table 4', we find that the D_1 values for the new estimator are much less than that of transformation method when n is large. And also the values of D_2 point out similar performance. We notice that the D_2 values for the new estimator are smaller than that of transformation method no matter whether n is large or not. When a density has a relatively heavy tail, the performance of the new estimator appears to be much better than the transformation method

by comparing the six parts of tables. As a result, we can say that the new method is better than the transformation method not only when estimating a density on the whole domain, but also on the tails.

This is further illustrated using the local comparison described further.

4.3 Local Comparison of the Two Estimators

For the local comparison, I plot the average squared errors based on the 100 replications for each density, i.e., I plot the values of

$$ASE(x) = \frac{1}{100}\sum_{i=1}^{100}\{\hat{f}_{[i]}(x) - f(x)\}^2, \tag{4.5}$$

where $\hat{f}_{[i]}(x)$ is the density estimator of $f(x)$ based on the i^{th} replication.

"A graph is worth a thousand words," and this is very well depicted in the present case. The graphs can explain the tail estimation more clearly and vividly.

These graphs are for the sample size $n = 100$ for the six models studied in this thesis. The solid line represents the performance of the new estimator and the dash line stands for the transformation method.

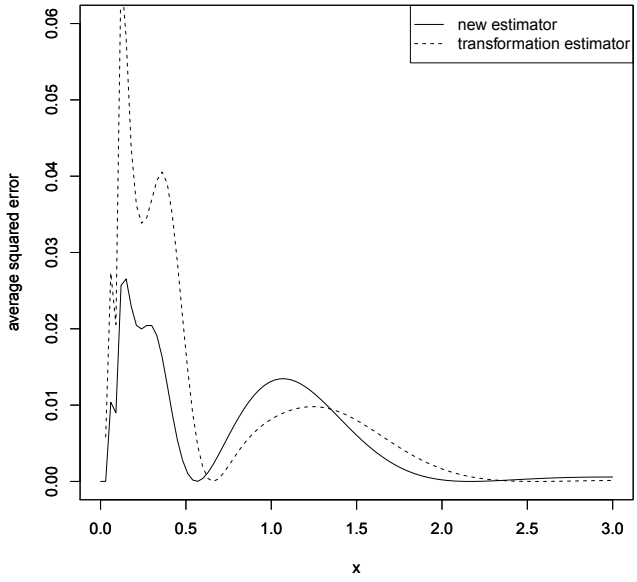

Figure 4.2: Average integrated squared error for Weibull distribution(c=1.5).

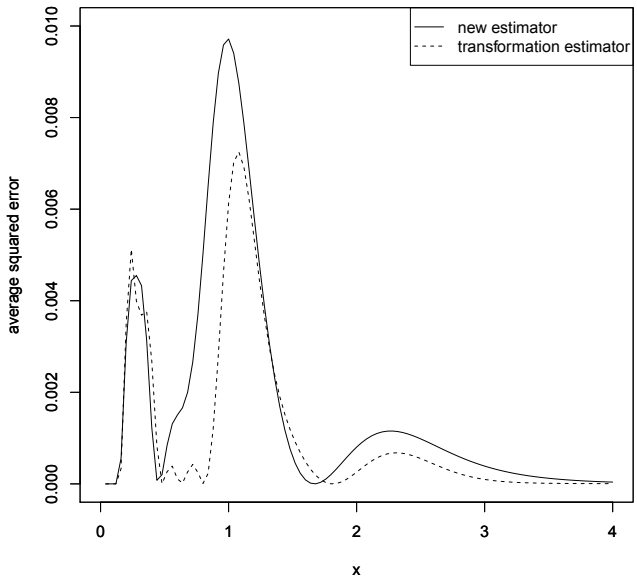

Figure 4.3: Average integrated squared error for lognormal distribution($\sigma = 0.5$).

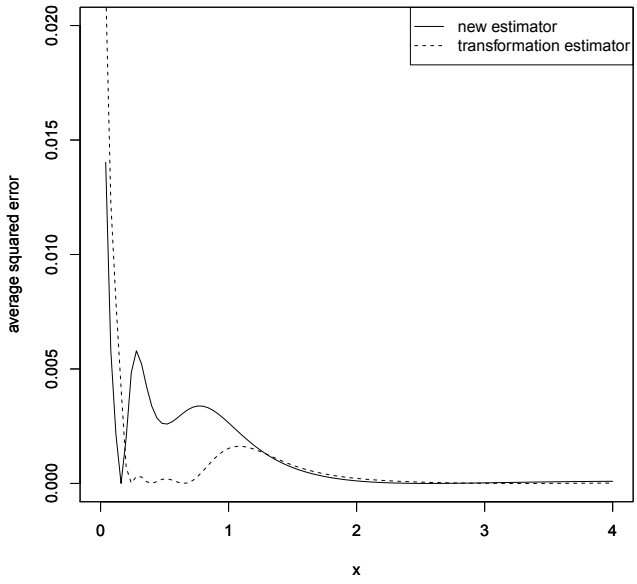

Figure 4.4: Average integrated squared error for lognormal distribution($\sigma = 1$).

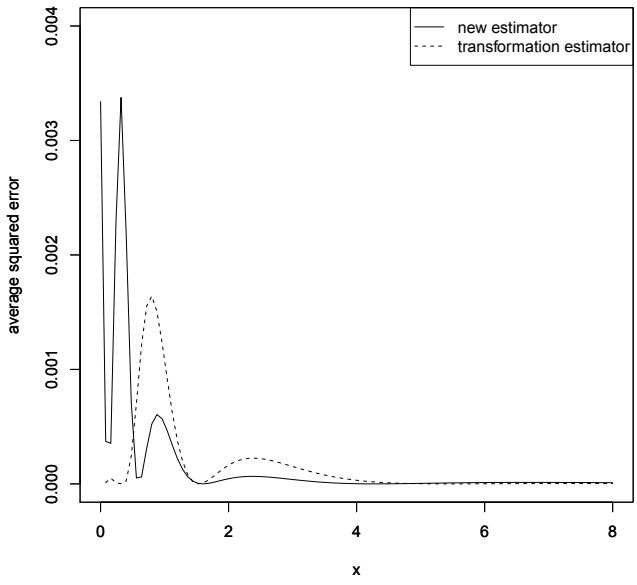

Figure 4.5: Average integrated squared error for mixture of 30% Pareto and 70% lognormal.

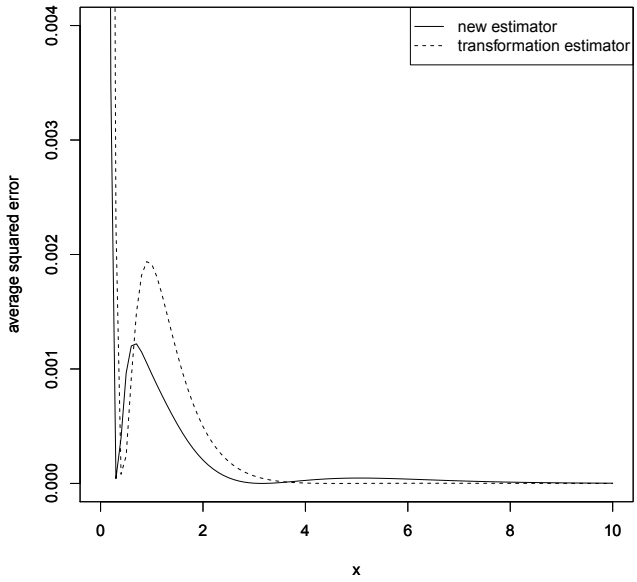

Figure 4.6: Average integrated squared error for mixture of 60% Pareto and 40% lognormal.

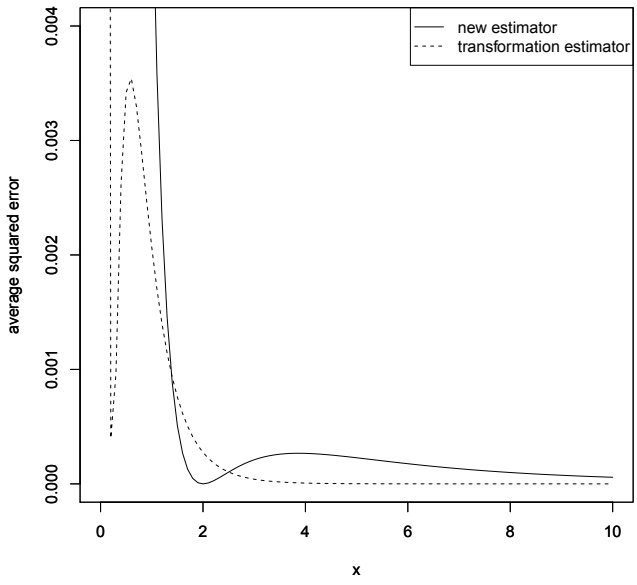

Figure 4.7: Average integrated squared error for mixture of 90% Pareto and 10% lognormal.

Focussing on the tails of the distribution, we see from Figure 4.1 that the new estimator is pretty much similar to the transformation estimator in terms of $ASE(x)$ when estimating the tail. However, in general we can say that the new method prosed by Chaubey et al. (2007) performs better than the transformation method on the whole domain for the Weibull distribution. Further looking at the Figure 4.2, the transformation estimator seems better than the new estimator when estimating the tail, however, only slightly, this being true on the whole support in this case. Figure 4.3 is for another lognormal distribution with parameter 1, but a larger variance and hence more heavier tail than the previous lognormal distribution. Here again, for estimating the tail, the performance of both the methods is very similar to each other but the new estimator seems better than the transformation method overall. Figures 4.4, 4.5 and 4.6 reflect the situation of the mixture loss models. We can clearly see that the tail estimation for both of the methods is pretty much the same in the cases of mixture models 1 and 2. While on the whole domain, we can say that the new estimator seems a little better than the transformation estimator for the first two mixture models. For the third mixture model, the transformation estimator seems better than the new estimator, especially in the lower tail of the distribution. To sum it up, the two methods are very comparable when estimating the loss models and the new method is relatively computationally more efficient than the transformation method and the transformation method does not offer any significant advantage.

4.4 Conclusion

When actuaries analyze loss models, they are always striving for a more accurate and easier method. Both of the two methods are quite good on estimating distributions describing

actuarial loss because they both alleviate the boundary problem and provide a meaningful description of the loss. However the transformation method is too complicated to implement without offering any significant advantage in terms of local or global comparison. Although they just simply use rule-of-thumb to determine the bandwidth in the paper Bolance et al. (2002), it may have to choose the transformation function in order to get a symmetric density. On the other hand, the new method proposed by Chaubey et al. (2007), performs pretty much as good as the transformation method in estimating some loss models, and may show better performance in some cases. For actuarial analysis of loss models, the tail estimation is more important. According to the simulation results based on global as well as local comparisons, we have enough evidence to conclude that the new method performs better or is qualitatively comparable to the transformation method, especially for the tail estimation. Also we note that the new method is computationally much simpler than the transformation method because we don't need extra computation time to choose the optimal transformation.

In order to facilitate the use of nonparametric density estimation, we always look to make things easier and more approachable. The new method proposed by Chaubey et al. (2007) presents such a method which adapts the kernel method for the whole real line to the non-negative data by using asymmetric kernels. It provides the actuaries a new perspective in order to estimate loss models non-perimetrically that is computationally very efficient and seems as good as competing methods available in the literature. Overall, it is good for us to have many different estimators to use and each method has its own pros and cons, and any decision maker has to find a balance between the computational efficiency and accuracy of procedure. The new density estimator seems to provide such a balance for estimating loss models.

References

[1] Altman, N. and Leger, C. (1994). Bandwidth selection for kernel distribution function estimation. *Journal of Statistical Planning and Inference,* **46***,* 195-214.

[2] Bolance, C. , Guillen, M. and Nielsen, J. P. (2002). Kernel density estimation of actuarial loss functions. *Insurance: Mathematics and Economics,* **32***,* 19-16.

[3] Bowman, A. W. (1984). An Alternative Method of Cross-Validation for the Smoothing of Density Estimates. *Biometrika,* **71***,* 353-360.

[4] Buch-Larsen, T., Nielsen, J. P., Guillen, M. and Bolance, C. (2005). Kernel density estimation for heavy-tailed distributions using the champernowne transformation. *Statistics,* **39***,* 503-518.

[5] Chaubey, Y. P., Sen, A. and Sen, P. K. (2007). A new smooth density estimator for non-negative random variables. *Technical report, Dept. of Mathematics and Statistics, Concordia University, Canada.*

[6] Chaubey, Y. P., and Sen, P. K. (1996). One smooth estimation of survival and density functions. *Statist. Decisions,* **14***,* 1-22.

[7] Deheuvel, P. (1977). Estimation non parametrique de la densite par histogrammes generalises(II), *Pub. Inst. Stat. Univ. Paris,* **XXII***,* 1-23.

[8] Feller, W. (1965). *An introduction to probability theory and its application* **(Vol, II)***,* New York: John Wiley and Sons.

[9] Hall, P. and Marron, J. S. (1987). Estimation of integrated squared density derivatives. *Statistics and Probability letters,* **6***,* 109-115.

[10] Hall, P. and Marron, J. S. (1990). Using the bootstrap to estimate mean squared error and select smoothing parameter in nonparametric problems. *Journal of Multivariate Analysis,* **32***,* 177-203.

[11] Hardle, W. (1990). *Smoothing Techniques: With Implementation in S.* Springer.

[12] Izenman, A. J. (1991). Recent Developments in Nonparametric density estimation. *Journal of the American Statistical Association,* **86***,* 205-224.

[13] Jones, O., Maillardet, R. and Robinson, A. (2009). *Introduction to scientific programming and simulation using R.* Chapman and Hall.

[14] Klugman, S. A., Panjer, H. H. and Willmot, G. E. (2008). *Loss Models: from data to decisions (3rd ed).* Hoboken, N.J.

[15] Loader, C. R. (1999). Bandwidth Selection: Classical or Plug-in? *The Annals of Statistics,* **27***,* 415-48.

[16] Mugdadi, A. and Jetter, J. (2010). A simulation study for the bandwidth selection in the kernel density estimation based on the exact and the asymptotic MISE. *Pak. J. Statist,* **26** 239-265.

[17] Park, B. U. and Marron, J. S. (1990). Comparison of data driven bandwidth selectors. *J. Amer. Statist. Assoc.,* **85***,* 66-72.

[18] Parzen, E. (1962). On estimation of a probability density function and mode. *Annals of Mathematical Statistics,* **33***,* 1065-1076.

[19] Raykar, V. C. and Duraiswami, R. (2006). Fast optimal bandwidth selection for kernel density estimation. *Preprint, Dept. of computer science and UMIACS, University of Maryland, CollegePark.*

[20] Rosenblatt, M. (1956). Remarks on some nonparametric estimates of a density function. *Annals of Mathematical Statistics,* **27***,* 832-837.

[21] Rudemo, M. (1982). Empirical Choice of Histograms and Kernel Density Estimators. *Scandinavian Journal of Statistics,* **9***,* 65-78.

[22] Scott, D. W. and Terrell, G. R. (1987). Biased and Unbiased Cross-Validation in Density

Estimation. *Journal of the American Statistical Association,* **82***,* 1131-1146.

[23] Silverman, B. W. (1986). *Density Esitmation for Statistics and Data Analysis.* Chapman and Hall.

[24] Terrell, G. R. (1990). The Maximal Smoothing Principle in Density Estimation. *Journal of the American Statistical Association,* **85***,* 470-477.

[25] Turlach, B. A. (1993). Bandwidth selection in kernel density estimation: A review. *CORE and Institute de Statistique.*

[26] Wand, M. P. and Jones, M. C. (1995). *Kernel Smoothing.* Chapman and Hall.

[27] Wand, M. P., Marron, J. S. and Ruppert, D. (1991). Transformation in Density Estimation. *Journal of the American Statistics Association,* **86***,* 343-353.

[28] Wang, B. and Wang, X. F. (2007). Bandwidth selection for weighted kernel density estimation. *Electronic Journal of Statistics,* **0***,* 1935-7524.

Appendix A: R-codes for Computing D_1 and D_2 for the Transformation Method (An example for Weibull distribution with c=1.5)

```
#Parameters selection

fn1=function(y,r){

n=length(y)

r1=r[1]

r2=r[2]

z=rep(0,n)

for (i1 in 1:n){

if(r2==0){z[i1]=log(y[i1]+r1)}

else{z[i1]=(y[i1]+r1)^r2}

}

return(z)

}

fn11=function(y,r){

n=length(y)

r1=r[1]

r2=r[2]

s=0

for (i in 1:(n-1)){

for (j in (i+1):n){

if(r2==0){
```

```
s=s+(sqrt(mean((fn1(y,r)-mean(fn1(y,r)))^2))*((21/(40*sqrt(2)*n^2)))^(1/13))^(-5
(dnorm(1/(sqrt(mean((fn1(y,r)-mean(fn1(y,r)))^2))*((21/(40*sqrt(2)*n^2)))^(1/13)
(log(y[i]+r1)-log(y[j]+r1))))^2}
else{s=s+(sqrt(mean((fn1(y,r)-mean(fn1(y,r)))^2))*((21/(40*sqrt(2)*n^2)))^(1/13)
(-5)*(dnorm(1/(sqrt(mean((fn1(y,r)-mean(fn1(y,r)))^2))*((21/(40*sqrt(2)*n^2)))^
(1/13))*((y[i]+r1)^r2-(y[j]+r1)^r2)))^2}
}
}
s=s/(n*(n-1))
return(s)
}
bandwidth=function(y,r){
n=length(y)
r1=r[1]
r2=r[2]
if(r2==0){b=1.059*sqrt(mean((log(y+r1)-mean(log(y+r1)))^2))*n^(-1/5)}
else{b=1.059*sqrt(mean(((y+r1)^r2-mean((y+r1)^r2))^2))*n^(-1/5)}
return(b)
}
fn2=function(x,y,r,h){
n=length(x)
r1=r[1]
r2=r[2]
f=rep(0,n)
```

```
for (i in 1:n){
if(r2==0){
f[i]=(mean(dnorm((log((y+r1)/(x[i]+r1)))/h))/(h*(x[i]+r1))-dweibull(x[i],1.5))^2
else{
f[i]=(mean(dnorm((((y+r1)^r2-(x[i]+r1)^r2)/h)))/h*(r2*(x[i]+r1)^(r2-1))-
dweibull(x[i],1.5))^2}
}
return(f)
}
fn3=function(x,y,r,h){
n=length(x)
r1=r[1]
r2=r[2]
k=rep(0,n)
for (i in 1:n){
if(r2==0){
k[i]=(x[i]^2*mean(dnorm((log((y+r1)/(x[i]+r1)))/h))/(h*(x[i]+r1))-x[i]^2*
dweibull(x[i],1.5))^2}
else{
k[i]=(x[i]^2*mean(dnorm((((y+r1)^r2-(x[i]+r1)^r2)/h)))/h*(r2*(x[i]+r1)^(r2-1))-x
dweibull(x[i],1.5))^2}
}
return(k)
}
```

```
n=100

times=100

M1=rep(0,times)

M2=rep(0,times)

for (i in 1:times){

set.seed(1+i*times)

y=rweibull(n,1.5)

a=optim(c(0, 0),fn11,y=y,lower=c(0,100),upper=c(-100,1),method ="L-BFGS-B")

r=a$par

h=bandwidth(y,r)

}


#Comparison of D1 and D2

bandwidth=function(y){

n=length(y)

b=1.059*sqrt(mean((log(y)-mean(log(y)))^2))*n^(-1/5)

return(b)

}

fn2=function(x,y,h){

n=length(x)

f=rep(0,n)

for (i in 1:n){

f[i]=(mean(dnorm((log((y)/(x[i])))/h))/(h*(x[i]))-dweibull(x[i],1.5))^2

}
```

```
return(f)
}
fn3=function(x,y,h){
n=length(x)
k=rep(0,n)
for (i in 1:n){
k[i]=(x[i]^2*mean(dnorm((log((y)/(x[i])))/h))/(h*(x[i]))-x[i]^2*dweibull(x[i],1
}
return(k)
}
n=100
times=100
M1=rep(0,times)
M2=rep(0,times)
for (i in 1:times){
set.seed(1+i*times)
y=rweibull(n,1.5)
h=bandwidth(y)
M1[i]<-mean(fn2(y,y,h))
M2[i]<-mean(fn3(y,y,h))}
```

Appendix B: R-codes for Computing D_1 and D_2 for the New Method (An example for Weibull distribution with c=1.5)

```
fn1<-function(x,y,h){

n1<-length(y)

h1<-h[1]

h2<-h[2]

f1<-0

f2<-0

f3<-0

f4<-0

for (j in 1:n1){

l<-dgamma(y[j]/(x+h2),h1,h1)

f1<-f1+y[j]*l

f2<-f2+((y[j])^2)*l

f3<-f3+((y[j])^3)*l

f4<-f4+y[j]*dgamma(y[j]/h2,h1,h1)

}

f1<-f1/n1;f2<-f2/n1;f3<-f3/n1;f4<-f4/n1/(1/h2^2);

g1<-sqrt(h1)*(f1)/(((x+h2)^3)*sqrt(4*pi)*n1)+((h2+x/h1)*(h1/(x+h2)^4*f2-(h1+1)/

(x+h2)^3*f1)+x^2/(2*h1)*(h1^2/(x+h2)^6*f3-(2*h1^2+4*h1)/(x+h2)^5*f2+

(h1^2+3*h1+2)/(x+h2)^4*f1)+f4*h2*f1/(x+h2)^2)^2

return(g1)
```

```
}
bcv<-function(y,h){
h1<-h[1]
h2<-h[2]
a<-integrate(fn1,lower=0,upper=Inf,y=y,h=h,abs.tol=0.1^100)
b<-a$value
return(b)
}
fn2<-function(x,y,h){
n2<-length(x)
h1<-h[1]
h2<-h[2]
f<-rep(0,n2)
for (i1 in 1:n2){
f[i1]<-(mean(dgamma(y/(x[i1]+h2),h1,h1)*y)/(((x[i1]+h2)^2)*mean(pgamma(y/h2,h1,h
-dweibull(x[i1],1.5))^2
}
return(f)
}
fn3<-function(x,y,h){
n3<-length(x)
h1<-h[1]
h2<-h[2]
k<-rep(0,n3)
```

```
for (i2 in 1:n3){
k[i2]<-(mean(dgamma(y/(x[i2]+h2),h1,h1)*y)/(((x[i2]+h2)^2)*mean(pgamma
(y/h2,h1,h1)))*x[i2]^2-dweibull(x[i2],1.5))^2*x[i2]^2
}
return(k)
}
n<-100
times<-100
M1<-rep(0,times)
M2<-rep(0,times)
for (i in 1:times){
set.seed(i+i*times)
y<-rweibull(n,1.5)
d<-optim(c(3,0.01),bcv,y=y,lower=c(1,0.1^(200)),upper=c(200,1),method ="L-BFGS-B
h<-d$par
if(h[2]<0) h[2]=0
M1[i]<-mean(fn2(y,y,h))
M2[i]<-mean(fn3(y,y,h))
}
```